

# Curso de L<sup>A</sup>T<sub>E</sub>X

27 de febrero de 2025

## 1 Introducción

## 2 Estructura Básica

- Documento Mínimo
- Paquetes
- Entornos
- Dando Formato al texto

## 3 Listas

## 4 Secciones

# ¿Qué es L<sup>A</sup>T<sub>E</sub>X?

- L<sup>A</sup>T<sub>E</sub>X es un sistema de preparación de documentos.

# ¿Qué es L<sup>A</sup>T<sub>E</sub>X?

- L<sup>A</sup>T<sub>E</sub>X es un sistema de preparación de documentos.
- Utilizado para la creación de documentos científicos y técnicos.

# ¿Qué se necesita?

- A diferencia de muchos programas informáticos,  $\text{\LaTeX}$  no es una única aplicación que lo “contenga todo” en un solo lugar.

# ¿Qué se necesita?

- A diferencia de muchos programas informáticos,  $\text{\LaTeX}$  no es una única aplicación que lo “contenga todo” en un solo lugar.
- En cambio, consta de programas separados que trabajan en conjunto.

# ¿Qué se necesita?

- A diferencia de muchos programas informáticos,  $\text{\LaTeX}$  no es una única aplicación que lo “contenga todo” en un solo lugar.
- En cambio, consta de programas separados que trabajan en conjunto.
- Podemos dividirlos en dos elementos que realmente se necesitan:

# ¿Qué se necesita?

- A diferencia de muchos programas informáticos,  $\text{\LaTeX}$  no es una única aplicación que lo “contenga todo” en un solo lugar.
- En cambio, consta de programas separados que trabajan en conjunto.
- Podemos dividirlos en dos elementos que realmente se necesitan:
  - Un sistema TeX.



# ¿Qué se necesita?

- A diferencia de muchos programas informáticos,  $\text{\LaTeX}$  no es una única aplicación que lo “contenga todo” en un solo lugar.
- En cambio, consta de programas separados que trabajan en conjunto.
- Podemos dividirlos en dos elementos que realmente se necesitan:
  - Un sistema TeX.
  - Un editor de texto.

- El núcleo del trabajo con  $\text{\LaTeX}$  es tener disponible un sistema TeX.

- El núcleo del trabajo con  $\text{\LaTeX}$  es tener disponible un sistema TeX.
- Un sistema TeX es un conjunto de programas y archivos necesarios para que  $\text{\LaTeX}$  funcione.

- El núcleo del trabajo con L<sup>A</sup>T<sub>E</sub>X es tener disponible un sistema TeX.
- Un sistema TeX es un conjunto de programas y archivos necesarios para que L<sup>A</sup>T<sub>E</sub>X funcione.
- Existen dos sistemas TeX principales: MiKTeX y TeX Live. Ambos disponibles para Windows, macOS y Linux.

- El núcleo del trabajo con L<sup>A</sup>T<sub>E</sub>X es tener disponible un sistema TeX.
- Un sistema TeX es un conjunto de programas y archivos necesarios para que L<sup>A</sup>T<sub>E</sub>X funcione.
- Existen dos sistemas TeX principales: MiKTeX y TeX Live. Ambos disponibles para Windows, macOS y Linux.
- MiKTeX tiene un fuerte respaldo en Windows; en macOS, TeX Live está incluido en una colección más grande llamada MacTeX.

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con LaTeX, ya que ofrecen funciones como:

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con LaTeX, ya que ofrecen funciones como:
  - Compilación de archivos con un solo clic.



# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con LaTeX, ya que ofrecen funciones como:
  - Compilación de archivos con un solo clic.
  - Visores de PDF integrados.

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con LaTeX, ya que ofrecen funciones como:
  - Compilación de archivos con un solo clic.
  - Visores de PDF integrados.
  - Resaltado de sintaxis.

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con LaTeX, ya que ofrecen funciones como:
  - Compilación de archivos con un solo clic.
  - Visores de PDF integrados.
  - Resaltado de sintaxis.
- Existen muchos editores de  $\text{\LaTeX}$ , entre los que podemos enumerar.

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con  $\text{\LaTeX}$ , ya que ofrecen funciones como:
  - Compilación de archivos con un solo clic.
  - Visores de PDF integrados.
  - Resaltado de sintaxis.
- Existen muchos editores de  $\text{\LaTeX}$ , entre los que podemos enumerar.
  - TeXworks, está incluido en TeX Live y MiKTeX para Windows y Linux

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con  $\text{\LaTeX}$ , ya que ofrecen funciones como:
  - Compilación de archivos con un solo clic.
  - Visores de PDF integrados.
  - Resaltado de sintaxis.
- Existen muchos editores de  $\text{\LaTeX}$ , entre los que podemos enumerar.
  - TeXworks, está incluido en TeX Live y MiKTeX para Windows y Linux
  - TeXShop, incluido en MacTeX.

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con  $\text{\LaTeX}$ , ya que ofrecen funciones como:
  - Compilación de archivos con un solo clic.
  - Visores de PDF integrados.
  - Resaltado de sintaxis.
- Existen muchos editores de  $\text{\LaTeX}$ , entre los que podemos enumerar.
  - TeXworks, está incluido en TeX Live y MiKTeX para Windows y Linux
  - TeXShop, incluido en MacTeX.
  - Winedt, un editor comercial para Windows.

# Editor de Texto

- Los archivos de  $\text{\LaTeX}$  son archivos de texto plano con extensión `.tex`, por lo que pueden editarse con cualquier editor de texto.
- Sin embargo, es conveniente utilizar un editor diseñado para trabajar con  $\text{\LaTeX}$ , ya que ofrecen funciones como:
  - Compilación de archivos con un solo clic.
  - Visores de PDF integrados.
  - Resaltado de sintaxis.
- Existen muchos editores de  $\text{\LaTeX}$ , entre los que podemos enumerar.
  - TeXworks, está incluido en TeX Live y MiKTeX para Windows y Linux
  - TeXShop, incluido en MacTeX.
  - Winedt, un editor comercial para Windows.
  - Overleaf, un editor en línea.



Figura: Distintos editores de L<sup>A</sup>T<sub>E</sub>X.



# Documento Mínimo

- La estructura básica de un documento es la siguiente:

```
\documentclass{article}  
\begin{document}  
    Hello World!  
\end{document}
```

# Documento Mínimo

- La estructura básica de un documento es la siguiente:

```
\documentclass{article}  
\begin{document}  
    Hello World!  
\end{document}
```

- El comando `\documentclass` indica el tipo de documento que se va a crear.

# Documento Mínimo

- La estructura básica de un documento es la siguiente:

```
\documentclass{article}  
\begin{document}  
    Hello World!  
\end{document}
```

- El comando `\documentclass` indica el tipo de documento que se va a crear.
- El *argumento* en llaves `{ }` le dice a  $\text{\LaTeX}$  qué tipo de documento estamos creando: en este ejemplo, `article`.

# Documento Mínimo

- La estructura básica de un documento es la siguiente:

```
\documentclass{article}
\begin{document}
    Hello World!
\end{document}
```

- El comando `\documentclass` indica el tipo de documento que se va a crear.
- El *argumento* en llaves `{ }` le dice a L<sup>A</sup>T<sub>E</sub>X qué tipo de documento estamos creando: en este ejemplo, `article`.
- Un signo de porcentaje `%` comienza un *comentario* — L<sup>A</sup>T<sub>E</sub>X ignorará el resto de la línea.

# `\documentclass`

- `\documentclass` es un comando que le dice a  $\text{\LaTeX}$  qué tipo de documento estamos creando.

# `\documentclass`

- `\documentclass` es un comando que le dice a  $\text{\LaTeX}$  qué tipo de documento estamos creando.
- Algunos tipos de documentos comunes son:
  - `article`: Artículos de revistas, presentaciones, informes cortos, documentación, invitaciones, etc.

# `\documentclass`

- `\documentclass` es un comando que le dice a  $\text{\LaTeX}$  qué tipo de documento estamos creando.
- Algunos tipos de documentos comunes son:
  - `article`: Artículos de revistas, presentaciones, informes cortos, documentación, invitaciones, etc.
  - `report`: Informes más largos que contienen varios capítulos, libros pequeños, tesis, etc.

# `\documentclass`

- `\documentclass` es un comando que le dice a  $\text{\LaTeX}$  qué tipo de documento estamos creando.
- Algunos tipos de documentos comunes son:
  - `article`: Artículos de revistas, presentaciones, informes cortos, documentación, invitaciones, etc.
  - `report`: Informes más largos que contienen varios capítulos, libros pequeños, tesis, etc.
  - `book`: Libros.



# `\documentclass`

- `\documentclass` es un comando que le dice a  $\text{\LaTeX}$  qué tipo de documento estamos creando.
- Algunos tipos de documentos comunes son:
  - `article`: Artículos de revistas, presentaciones, informes cortos, documentación, invitaciones, etc.
  - `report`: Informes más largos que contienen varios capítulos, libros pequeños, tesis, etc.
  - `book`: Libros.
  - `letter`: Cartas.

- `\documentclass` es un comando que le dice a  $\text{\LaTeX}$  qué tipo de documento estamos creando.
- Algunos tipos de documentos comunes son:
  - `article`: Artículos de revistas, presentaciones, informes cortos, documentación, invitaciones, etc.
  - `report`: Informes más largos que contienen varios capítulos, libros pequeños, tesis, etc.
  - `book`: Libros.
  - `letter`: Cartas.
  - `beamer`: Presentaciones.

# `\documentclass`

- El comando `\documentclass` posee conjuntos de opciones que van entre corchetes `[ ]`. Algunas de ellas son:
  - `10pt`, `11pt`, `12pt`: Tamaño de la fuente.

# `\documentclass`

- El comando `\documentclass` posee conjuntos de opciones que van entre corchetes `[ ]`. Algunas de ellas son:
  - `10pt`, `11pt`, `12pt`: Tamaño de la fuente.
  - `a4paper`, `letterpaper`, `legalpaper`: Tamaño del papel.

# `\documentclass`

- El comando `\documentclass` posee conjuntos de opciones que van entre corchetes `[ ]`. Algunas de ellas son:
  - `10pt`, `11pt`, `12pt`: Tamaño de la fuente.
  - `a4paper`, `letterpaper`, `legalpaper`: Tamaño del papel.
  - `twocolumn`: Dos columnas.

# `\documentclass`

- El comando `\documentclass` posee conjuntos de opciones que van entre corchetes `[ ]`. Algunas de ellas son:
  - `10pt`, `11pt`, `12pt`: Tamaño de la fuente.
  - `a4paper`, `letterpaper`, `legalpaper`: Tamaño del papel.
  - `twocolumn`: Dos columnas.
  - `twoside`, `oneside`: Impresión a doble o una cara.

# `\documentclass`

- El comando `\documentclass` posee conjuntos de opciones que van entre corchetes `[ ]`. Algunas de ellas son:
  - `10pt`, `11pt`, `12pt`: Tamaño de la fuente.
  - `a4paper`, `letterpaper`, `legalpaper`: Tamaño del papel.
  - `twocolumn`: Dos columnas.
  - `twoside`, `oneside`: Impresión a doble o una cara.
- Por ejemplo, `\documentclass[12pt,a4paper]{article}` indica que el documento será un artículo con fuente de 12 puntos y tamaño de papel A4.

- Los paquetes son archivos que contienen comandos y entornos adicionales para L<sup>A</sup>T<sub>E</sub>X.



- Los paquetes son archivos que contienen comandos y entornos adicionales para L<sup>A</sup>T<sub>E</sub>X.
- Se cargan en el preámbulo del documento después del comando `\documentclass`.

- Los paquetes son archivos que contienen comandos y entornos adicionales para L<sup>A</sup>T<sub>E</sub>X.
- Se cargan en el preámbulo del documento después del comando `\documentclass`.
- El comando `\usepackage[]{}{}` permite cargar un complemento (plugin), que añade nuevas funcionalidades.

- Existen numerosos complementos (por ejemplo, para mostrar imágenes, crear tablas, dibujar fórmulas químicas, generar cuadrículas de sudoku, etc.).

Ejemplos:

- Existen numerosos complementos (por ejemplo, para mostrar imágenes, crear tablas, dibujar fórmulas químicas, generar cuadrículas de sudoku, etc.).

## Ejemplos:

- `\usepackage[utf8]{inputenc}`: Carga el paquete inputenc con la opción utf8 (esto es para la codificación de caracteres).

- Existen numerosos complementos (por ejemplo, para mostrar imágenes, crear tablas, dibujar fórmulas químicas, generar cuadrículas de sudoku, etc.).

## Ejemplos:

- `\usepackage[utf8]{inputenc}`: Carga el paquete `inputenc` con la opción `utf8` (esto es para la codificación de caracteres).
- `\usepackage[T1]{fontenc}`: Especifica que se está utilizando el paquete de fuentes `T1`.

- Existen numerosos complementos (por ejemplo, para mostrar imágenes, crear tablas, dibujar fórmulas químicas, generar cuadrículas de sudoku, etc.).

## Ejemplos:

- `\usepackage[utf8]{inputenc}`: Carga el paquete inputenc con la opción utf8 (esto es para la codificación de caracteres).
- `\usepackage[T1]{fontenc}`: Especifica que se está utilizando el paquete de fuentes T1.
- `\usepackage[spanish]{babel}`: Carga el paquete babel, que se encarga de la tipografía con el idioma español.

- Existen numerosos complementos (por ejemplo, para mostrar imágenes, crear tablas, dibujar fórmulas químicas, generar cuadrículas de sudoku, etc.).

## Ejemplos:

- `\usepackage[utf8]{inputenc}`: Carga el paquete inputenc con la opción utf8 (esto es para la codificación de caracteres).
- `\usepackage[T1]{fontenc}`: Especifica que se está utilizando el paquete de fuentes T1.
- `\usepackage[spanish]{babel}`: Carga el paquete babel, que se encarga de la tipografía con el idioma español.
- `\usepackage{graphicx}`: Carga el paquete que permite incluir imágenes externas en el documento.

# Entornos

- Los entornos definen un “bloque”: todo el texto dentro de este bloque se transformará según la función del entorno.



# Entornos

- Los entornos definen un “bloque”: todo el texto dentro de este bloque se transformará según la función del entorno.
- Un entorno siempre comienza con `\begin{}` y termina con `\end{}`. Dentro de las `{ }` se especifica el nombre del entorno.

# Entornos

- Los entornos definen un “bloque”: todo el texto dentro de este bloque se transformará según la función del entorno.
- Un entorno siempre comienza con `\begin{}` y termina con `\end{}`. Dentro de las `{ }` se especifica el nombre del entorno.
- El entorno `document` es obligatorio: lo que está dentro constituye el contenido del documento. Fuera del bloque `document`, se encuentran comandos que modifican las características del documento o cómo se imprime (por ejemplo, paquetes o comandos globales).

# Entornos

- Los entornos definen un “bloque”: todo el texto dentro de este bloque se transformará según la función del entorno.
- Un entorno siempre comienza con `\begin{}` y termina con `\end{}`. Dentro de las `{ }` se especifica el nombre del entorno.
- El entorno `document` es obligatorio: lo que está dentro constituye el contenido del documento. Fuera del bloque `document`, se encuentran comandos que modifican las características del documento o cómo se imprime (por ejemplo, paquetes o comandos globales).
- Todos los demás entornos son opcionales y se usan según sea necesario.

**Ejemplo:** El entorno `itemize` crea listas con viñetas (listas sin numerar). Por lo tanto, una lista con viñetas se crea cada vez que se llama al comando `\item`.

# Dando Formato al texto

L<sup>A</sup>T<sub>E</sub>X tiene comandos para dar formato al texto.

- `\textbf{texto}`: **Texto en negrita.**

# Dando Formato al texto

L<sup>A</sup>T<sub>E</sub>X tiene comandos para dar formato al texto.

- `\textbf{texto}`: **Texto en negrita.**
- `\textit{texto}`: *Texto en cursiva.*

# Dando Formato al texto

L<sup>A</sup>T<sub>E</sub>X tiene comandos para dar formato al texto.

- `\textbf{texto}`: **Texto en negrita.**
- `\textit{texto}`: *Texto en cursiva.*
- `\underline{texto}`: Texto subrayado.

# Dando Formato al texto

L<sup>A</sup>T<sub>E</sub>X tiene comandos para dar formato al texto.

- `\textbf{texto}`: **Texto en negrita.**
- `\textit{texto}`: *Texto en cursiva.*
- `\underline{texto}`: Texto subrayado.
- `\texttt{texto}`: Texto en fuente de máquina de escribir.

# Dando Formato al texto

L<sup>A</sup>T<sub>E</sub>X tiene comandos para dar formato al texto.

- `\textbf{texto}`: **Texto en negrita.**
- `\textit{texto}`: *Texto en cursiva.*
- `\underline{texto}`: Texto subrayado.
- `\texttt{texto}`: Texto en fuente de máquina de escribir.
- `\color{nombre color}`: **Texto en color.**



# Dando Formato al texto

L<sup>A</sup>T<sub>E</sub>X tiene comandos para dar formato al texto.

- `\textbf{texto}`: **Texto en negrita.**
- `\textit{texto}`: *Texto en cursiva.*
- `\underline{texto}`: Texto subrayado.
- `\texttt{texto}`: Texto en fuente de máquina de escribir.
- `\color{nombre color}`: **Texto en color.**
- Formato de tamaño de fuente: `\tiny`, `\scriptsize`, `\footnotesize`, `\small`, `\normalsize`, `\large`, `\Large`, `\LARGE`, `\huge`, `\Huge`.

# Dando Formato al texto

L<sup>A</sup>T<sub>E</sub>X tiene comandos para dar formato al texto.

- `\textbf{texto}`: **Texto en negrita.**
- `\textit{texto}`: *Texto en cursiva.*
- `\underline{texto}`: Texto subrayado.
- `\texttt{texto}`: Texto en fuente de máquina de escribir.
- `\color{nombre color}`: **Texto en color.**
- Formato de tamaño de fuente: `\tiny`, `\scriptsize`, `\footnotesize`, `\small`, `\normalsize`, `\large`, `\Large`, `\LARGE`, `\huge`, `\Huge`.
- Forzar un salto de línea: `\\`.

# Dando Formato al texto

El entorno `itemize` crea listas con viñetas (listas sin numerar).

```
\begin{itemize}  
  \item Elemento 1  
  \item Elemento 2  
  \item Elemento 3  
\end{itemize}
```

- Elemento 1
- Elemento 2
- Elemento 3

# Dando Formato al texto

El entorno `enumerate` crea listas numeradas.

```
\begin{enumerate}  
\item Elemento 1  
\item Elemento 2  
\item Elemento 3  
\end{enumerate}
```

- 1 Elemento 1
- 2 Elemento 2
- 3 Elemento 3

LaTeX puede dividir/estructurar documentos en múltiples niveles jerárquicos, dependiendo del tipo de documento con el que se trabaje.

- `\section{texto}`: Sección.

LaTeX puede dividir/estructurar documentos en múltiples niveles jerárquicos, dependiendo del tipo de documento con el que se trabaje.

- `\section{texto}`: Sección.
- `\subsection{texto}`: Subsección.

LaTeX puede dividir/estructurar documentos en múltiples niveles jerárquicos, dependiendo del tipo de documento con el que se trabaje.

- `\section{texto}`: Sección.
- `\subsection{texto}`: Subsección.
- `\subsubsection{texto}`: Subsubsección.

LaTeX puede dividir/estructurar documentos en múltiples niveles jerárquicos, dependiendo del tipo de documento con el que se trabaje.

- `\section{texto}`: Sección.
- `\subsection{texto}`: Subsección.
- `\subsubsection{texto}`: Subsubsección.
- `\paragraph{texto}`: Párrafo.



LaTeX puede dividir/estructurar documentos en múltiples niveles jerárquicos, dependiendo del tipo de documento con el que se trabaje.

- `\section{texto}`: Sección.
- `\subsection{texto}`: Subsección.
- `\subsubsection{texto}`: Subsubsección.
- `\paragraph{texto}`: Párrafo.
- `\subparagraph{texto}`: Subpárrafo.

LaTeX puede dividir/estructurar documentos en múltiples niveles jerárquicos, dependiendo del tipo de documento con el que se trabaje.

- `\section{texto}`: Sección.
- `\subsection{texto}`: Subsección.
- `\subsubsection{texto}`: Subsubsección.
- `\paragraph{texto}`: Párrafo.
- `\subparagraph{texto}`: Subpárrafo.
- `\chapter{texto}`: Capítulo.

```
\documentclass{article}  
\usepackage[T1]{fontenc}  
\begin{document}  
  Hola Mundo!
```

Primer Documento.

```
\section{Primera Sección}  
Texto de la primera sección.
```

Segundo Párrafo.

```
\subsection{Subseccion de la primera sección}  
Texto de la subsección.
```

```
\subsubsection{Subsubsección de la primera sección}  
Texto de la subsubsección.
```

```
\end{document}
```

Hola Mundo!  
Primer Documento.

## 1 Primera Seccion

Texto de la seccion.  
Segundo parrafo.

### 1.1 Subseccion de la primera seccion

Texto de la subseccion.

#### 1.1.1 Subsubseccion de la primera seccion

Texto de la subsubseccion.

# Aclaraciones

- Las comillas son un poco complicadas: use el acento invertido ` sobre el lado izquierdo y el apóstrofe ' sobre el lado derecho.  
Comillas simple: ‘texto’.  
Comillas dobles: “texto”.

# Aclaraciones

- Las comillas son un poco complicadas: use el acento invertido ``` sobre el lado izquierdo y el apóstrofe `'` sobre el lado derecho.

Comillas simple: `'texto'`.

Comillas dobles: `"texto"`.

- Algunos caracteres comunes tienen significados especiales en  $\text{\LaTeX}$ :

`%` Signo de porcentaje

`#` Signo numeral

`&` Ampersand

`$` Signo pesos

# Aclaraciones

- Las comillas son un poco complicadas: use el acento invertido ``` sobre el lado izquierdo y el apóstrofe `'` sobre el lado derecho.

Comillas simple: `'texto'`.

Comillas dobles: `"texto"`.

- Algunos caracteres comunes tienen significados especiales en L<sup>A</sup>T<sub>E</sub>X:

`%` Signo de porcentaje

`#` Signo numeral

`&` Ampersand

`$` Signo pesos

- Si son usados, tendremos errores en la compilación. Para usar estos caracteres en la salida, se debe colocar barra invertida al caracter.

`\$ \% \& \#!`

- Los espacios en blanco en el código fuente de  $\text{\LaTeX}$  no tienen ningún efecto en el documento final.



# Aclaraciones

- Los espacios en blanco en el código fuente de  $\text{\LaTeX}$  no tienen ningún efecto en el documento final.
- $\text{\LaTeX}$  trata los espacios en blanco como “espacios en blanco”.

# Aclaraciones

- Los espacios en blanco en el código fuente de  $\text{\LaTeX}$  no tienen ningún efecto en el documento final.
- $\text{\LaTeX}$  trata los espacios en blanco como “espacios en blanco”.
- Para obtener un espacio en blanco en el documento final, se deben usar comandos especiales.

# Aclaraciones

- Los espacios en blanco en el código fuente de  $\text{\LaTeX}$  no tienen ningún efecto en el documento final.
- $\text{\LaTeX}$  trata los espacios en blanco como “espacios en blanco”.
- Para obtener un espacio en blanco en el documento final, se deben usar comandos especiales.
- Para obtener un espacio en blanco en el documento final, se deben usar comandos especiales.

# Tipografía Matemática

- ¿Por qué son especiales los signos pesos `\$`? Los usamos para marcar contenido matemático en el texto.

% no tan bueno:

Sean  $a$  y  $b$  tales que  $c = a - b + 1$ .

% mucho mejor:

Sean  $a$  y  $b$  tales que  $c = a - b + 1$ .

Sean  $a$  y  $b$  tales que  $c = a - b + 1$ .

Sean  $a$  y  $b$  tales que  $c = a - b + 1$ .

- Utilice siempre los signos de pesos en pares — uno para comenzar el contenido matemático, y uno para terminarlo.
- L<sup>A</sup>T<sub>E</sub>X maneja el espacio automáticamente; por lo que ignorará los que hayamos puesto.

Sea  $y=mx+b$  ...

Sea  $y = m x + b$  ...

Sea  $y = mx + b$  ...

Sea  $y = mx + b$  ...

# Tipografía Matemática:Notación

- Use el signo  $\wedge$  para indicar superíndices y el guión bajo  $_$  para marcar subíndices.

<code>\$y = c_2 x^2 + c_1 x + c_0\$</code>	$y = c_2 x^2 + c_1 x + c_0$
--	-----------------------------

- Utilice las llaves  $\{ \}$  para agrupar superíndices y subíndices.

<code>\$F_n = F_{n-1} + F_{n-2}\$% oops!</code>	$F_n = F_n - 1 + F_n - 2$
<code>\$F_n = F_{\{n-1\}} + F_{\{n-2\}}\$% ok!</code>	$F_n = F_{n-1} + F_{n-2}$

- Hay comandos para letras Griegas y notación común.

<code>\$\mu = A e^{\{Q/RT\}}\$</code>	$\mu = A e^{Q/RT}$
<code>\$\Omega = \sum_{\{k=1\}}^n \omega_k\$</code>	$\Omega = \sum_{k=1}^n \omega_k$

# Tipografía Matemática: Entornos

- `equation` es un *entorno*.
- Un comando puede producir diferentes salidas en diferentes contextos.

Podemos escribir `\Omega = \sum_{k=1}^n \omega_k` en nuestro texto, o podemos escribir `\begin{equation}`  
`\Omega = \sum_{k=1}^n \omega_k` `\end{equation}`  
para mostrarlo en un entorno diferente.

Podemos escribir  $\Omega = \sum_{k=1}^n \omega_k$  en nuestro texto, o podemos escribir

$$\Omega = \sum_{k=1}^n \omega_k \quad (1)$$

para mostrarlo en un entorno diferente.

- Note como el  $\Sigma$  es más grande en el entorno `equation`, y como el subíndice y superíndice cambian de posición, a pesar de que utilizamos los mismos comandos.

# Ejemplos con `amsmath`

- Utilice `equation*` (“ecuación-asterisco”) para ecuaciones no-numeradas.

```
\begin{equation*}
```

```
\Omega = \sum_{k=1}^n
```

```
\omega_k
```

```
\end{equation*}
```

$$\Omega = \sum_{k=1}^n \omega_k$$

- $\text{\LaTeX}$  trata las letras adyacentes como variables multiplicadas entre sí, lo cual no siempre es lo que se quiere. `amsmath` define comandos para muchos operadores matemáticos comunes.

```
\begin{equation*} \% \text{ bad!}
```

```
\min_{x,y} (1-x)^2
```

```
\end{equation*}
```

$$\min_{x,y} (1-x)^2$$

```
\begin{equation*} \% \text{ good!}
```

```
\min_{x,y} \{(1-x)^2\}
```

```
\end{equation*}
```

$$\min_{x,y} (1-x)^2$$